



# 18

## Nonparametric smoothing

---

Nonparametric smoothing or nonparametric function estimation grew enormously in the 1980s. The word ‘nonparametric’ has nothing to do with the classical rank-based methods, such as the Wilcoxon rank test, but it is understood as follows. The simplest relationship between an outcome  $y$  and a predictor  $x$  is given by a linear model

$$E(y) = \beta_0 + \beta_1 x.$$

This linear model is a parametric equation with two parameters  $\beta_0$  and  $\beta_1$ . A nonparametric model would simply specify  $E(y)$  as some function of  $x$

$$E(y) = f(x).$$

The class of all possible  $f(\cdot)$  is ‘nonparametric’ or infinite dimensional.

The literature on nonparametric smoothing is vast, and we cannot hope to do justice to it in a single chapter. We will focus on a general methodology that fits well with the likelihood-based mixed effects modelling. The approach is practical, treating functions with discrete rather than continuous index. This means it suffices to deal with the usual vectors and matrices, rather than function spaces and operators.

### 18.1 Motivation

**Example 18.1:** Figure 18.1 shows the scatter plot of SO<sub>2</sub> level and industrial activity; the latter is measured as the number of manufacturing enterprises employing 20 or more workers. In Section 6.8 we have shown that it is sensible to log-transform the data. Since our first instinct is that more industry leads to more pollution, when faced with this dataset, we might only consider a linear model (dotted line). A nonparametric regression estimate (solid line) suggests a quadratic model, shown in Section 6.8 to be well supported by the data. The nonparametric or quadratic fits are harder to interpret in this case, but in this empirical modelling there could be other confounding factors not accounted for by the variables. The idea is that we should let the data tell their story rather than impose our prejudice; with this attitude a nonparametric smoothing technique is an invaluable tool for exploratory data analysis. □

#### Ad hoc methods

Our general problem is as follows: given bivariate data  $(x_1, y_1), \dots, (x_N, y_N)$  we assume that conditional on  $x_i$  the outcome  $y_i$  is normal with mean

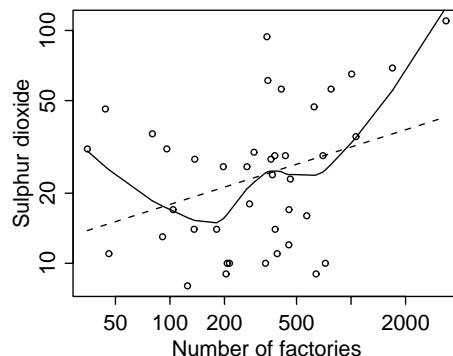


Figure 18.1: Relationship between  $SO_2$  level and industrial activity in 41 US cities. Shown are the nonparametric smooth (solid) and the linear regression (dashed) estimates.

$$E(y_i) = f(x_i)$$

and variance  $\sigma^2$ . We want to estimate the function  $f(x)$  from the data.

The key idea of nonparametric smoothing is *local averaging*:  $f(x)$  at a fixed value of  $x$  is the mean of  $y_i$  at  $x$ , so if there are many  $y_i$ 's observed at  $x$ , then the estimate of  $f(x)$  is the average of those  $y_i$ 's. More often than not we have to compromise: the estimate of  $f(x)$  is the average of  $y_i$ 's for  $x_i$ 's 'near'  $x$ . This can be implemented by partitioning the data, finding the nearest neighbours or kernel smoothing.

### Partitioning

We can partition the range of the predictor  $x$  into  $n$  small intervals or bins, so that within an interval  $f(x_i)$  is approximately constant, and  $y_i$ 's are approximately iid with mean  $f(x_i)$ . We can then estimate  $f(x_i)$  by the sample average of  $y_i$ 's in the corresponding interval. The estimated function can be drawn as the polygon connecting the sample means from each interval.

As an example, Table 18.1 partitions the  $SO_2$  data into 20 equispaced intervals (in  $\log x$ ). Note that some intervals are empty, but that does not affect the method. Figure 18.2 shows the nonparametric smoothing of the  $SO_2$  level against the industry using different numbers of bins. The amount of smoothing is determined by the interval size, which has the following trade-off: if the interval is too large then the estimate might smooth out important patterns in  $f(x)$ , and the estimate is biased; but if it is too small the noise variance exaggerates the local variation and obscures the real patterns. The purpose of smoothing is to achieve a balance between

No.	$x$	Bin	Mid- $x$	$y$	No.	$x$	Bin	Mid- $x$	$y$
1	35	1	39	31	22	361	11	383	28
2	44	2	49	46	23	368	11	383	24
3	46	2	49	11	24	379	11	383	29
4	80	4	78	36	25	381	11	383	14
5	91	5	98	13	26	391	11	383	11
6	96	5	98	31	27	412	11	383	56
7	104	5	98	17	28	434	12	482	29
8	125	6	123	8	29	453	12	482	12
9	136	6	123	14	30	454	12	482	17
10	137	6	123	28	31	462	12	482	23
11	181	8	193	14	32	569	13	605	16
12	197	8	193	26	33	625	13	605	47
13	204	8	193	9	34	641	13	605	9
14	207	8	193	10	35	699	14	760	29
15	213	8	193	10	36	721	14	760	10
16	266	9	243	26	37	775	14	760	56
17	275	10	305	18	38	1007	15	954	65
18	291	10	305	30	39	1064	15	954	35
19	337	10	305	10	40	1692	18	1891	69
20	343	11	383	94	41	3344	20	2984	110
21	347	11	383	61					

Table 18.1: Partitioning the  $SO_2$  level ( $= y$ ) data into 20 intervals/bins of the predictor variable  $x =$  industrial activities. ‘Mid- $x$ ’ is the midpoint (in log scale) of the interval. Note: throughout this section  $SO_2$  is analysed in log scale.

local bias and variance.

### Nearest neighbours

The *nearest-neighbour method* simply prescribes, for any  $x$ ,

$$\hat{f}(x) = \frac{1}{k} \sum_{i \in n_k(x)} y_i,$$

where  $n_k(x)$  is the neighbourhood of  $x$  that includes only the  $k$  values of  $x_i$ ’s nearest to  $x$ . Hence  $\hat{f}(x)$  is a simple average of  $y_i$ ’s for  $k$  nearest neighbours of  $x$ ; larger values of  $k$  effect more smoothing. For example, using  $k = 7$ , at  $x = 125$  we obtain the following nearest neighbours of  $x$  with the corresponding  $y$ :

$x$	125	136	137	104	96	91	181
$y$	8	14	28	17	31	13	14

giving an average log  $y$  of 2.79. The set of nearest neighbours needs to be computed at every value of  $x$ , making this method computationally more demanding than the partition method. For the plots in the top row of Figure 18.3  $\hat{f}(x)$  is computed at the observed  $x_i$ ’s, but this is not necessary as it can be computed at a smaller subset of values. Note that the

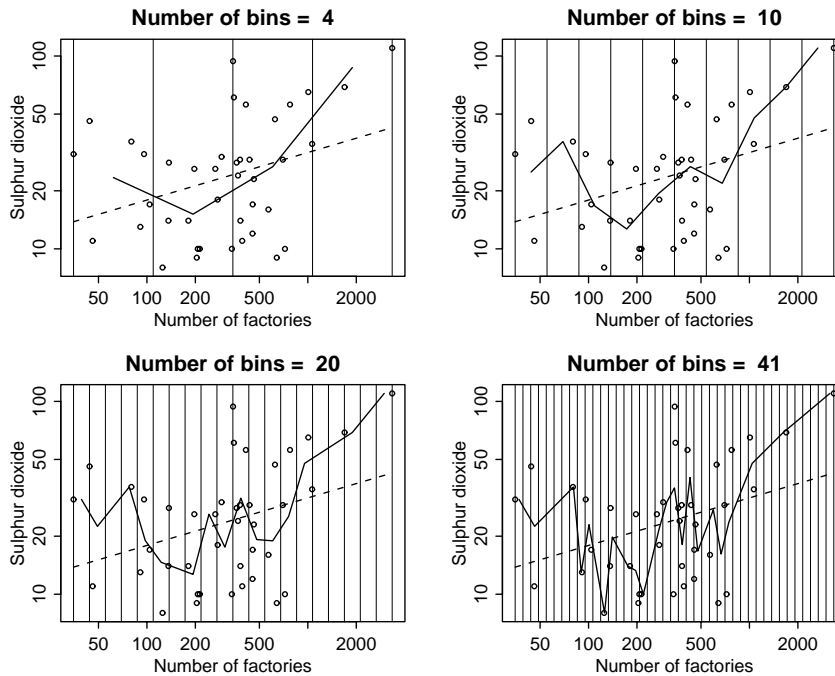


Figure 18.2: *Nonparametric smoothing of the  $SO_2$  level against the industrial activity using simple partitioning of the predictor variable. The dashed line in each plot is the linear regression line.*

estimate at the boundaries is biased, especially as we increase the number of neighbours.

### Kernel method

Using the *kernel method* one computes a weighted average

$$\hat{f}(x) = \frac{\sum_{i=1}^n k(x_i - x)y_i}{\sum_{i=1}^n k(x_i - x)}$$

where the kernel function  $k(x)$  is typically a symmetric density function. If we use the normal density the method is called Gaussian smoothing. The amount of smoothing is determined by the scale or width of the kernel; in Gaussian smoothing it is controlled by the standard deviation. The bottom row of Figure 18.3 shows the Gaussian smoothing of the  $SO_2$  data using a standard deviation of 0.2 and 0.05 (note:  $x$  is also in log scale).

With the last two methods  $\hat{f}(x)$  can be computed at any  $x$ , while with the first method the choice of a partition or interval size determines the values of  $x$  for which  $\hat{f}(x)$  is available. This is a weakness of the first method, since if we want  $f(x)$  for a lot of  $x$  values we have to make the intervals small, which in turn makes the estimation error large. The general

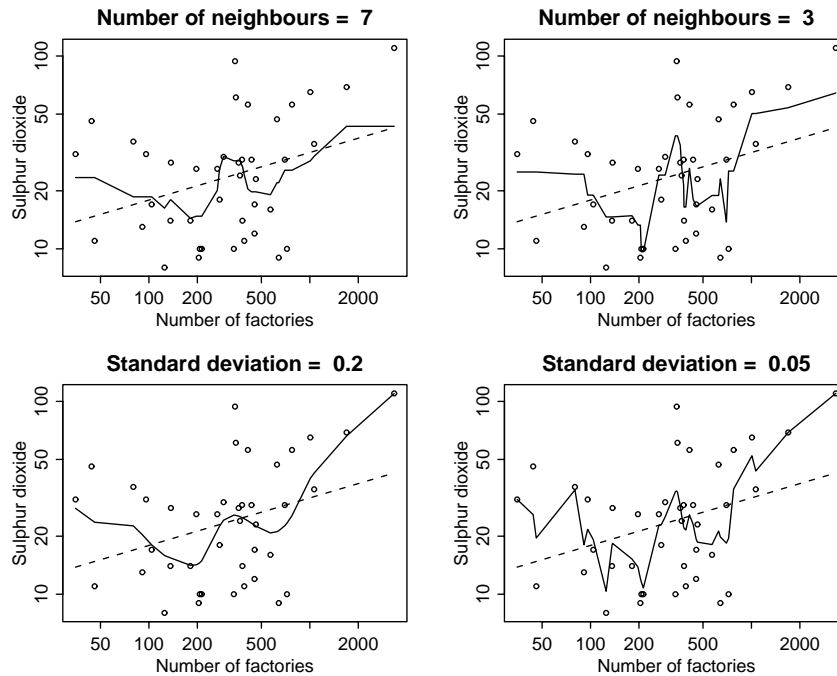


Figure 18.3: *Nonparametric smoothing of the  $SO_2$  level against the industrial activity using the nearest-neighbour method (top row) and the kernel method (bottom row). The kernel method uses the normal density with the stated standard deviation.*

method in the next section overcomes this weakness. For each method the choice of the smoothing parameter is an important issue. The next section shows that the problem is simply a variance component estimation problem.

## 18.2 Linear mixed models approach

We will now put the nonparametric function estimation within the linear mixed model framework with likelihood-based methodology. Compared to the nearest-neighbour or the kernel method, the likelihood-based method is easier to extend to deal with

- non-Gaussian outcome data, such as Poisson or binomial data;
- different types of functions, such as functions with jump discontinuities or partly parametric models;
- the so-called ‘inverse problems’ (e.g. O’Sullivan 1986): the observed data  $y$  satisfies a linear model  $Ey = X\beta$ , where  $\beta$  is a smooth function and  $X$  is ill-conditioned.
- higher-dimensional smoothing problems, including image analysis and disease mapping. The mixed model approach deals with the boundary

estimation automatically without any special handling. This feature is essential when we are dealing with higher-dimensional smoothing with irregular boundaries as in a geographical map, where the application of the kernel method is not straightforward.

While it is possible to develop the theory where  $x_i$ 's are not assumed to be equispaced (see Section 18.9), the presentation is simpler if we pre-bin the data in terms of the  $x$  values prior to analysis. So assume the  $x$  values form a regular grid; each  $x_i$  can be associated with several  $y$ -data, or perhaps none. This is exactly the same as partitioning the data as described before. Rather than just presenting the simple averages, the partition will be processed further. Pre-binning is commonly done in practice to reduce data volume, especially as the  $y$ -data at each  $x$  value may be summarized further into a few statistics such as the sample size, mean and variance. In many applications, such as time series or image analysis, the data usually come in a regular grid format.

The effect of binning is determined by the bin size: if it is too large then we introduce bias and lose some resolution of the original data, and in the limit as the bin size goes to zero we resolve the original data. In practice we make the bins small enough to preserve the original data (i.e. minimize bias and make local variation dominate), but large enough to be practical since there is a computational price for setting too many bins. We will not develop any theory to say how small is 'small enough', since in practice it is easy to recognize a large local variation, and if we are in doubt we can simply set it smaller. As a guideline, the degrees of freedom of the estimate (described in Section 18.5) should be much smaller than the number of bins.

So, after pre-binning, our problem is as follows. Given the observations  $(x_i, y_{ij})$  for  $i = 1, \dots, n$  and  $j = 1, \dots, n_i$ , where  $x_i$ 's are equispaced, we assume that  $y_{ij}$ 's are normal with mean

$$E(y_{ij}) = f(x_i) = f_i$$

and variance  $\sigma^2$ . We want to estimate  $f = (f_1, \dots, f_n)$ . Note that some  $n_i$ 's may be zero. Smoothness or other properties of  $f(x)$  will be imposed via some stochastic structure on  $f_i$ ; this is discussed in the next section.

To put this in the linear mixed model framework, first stack the data  $y_{ij}$ 's into a column vector  $y$ . Conditional on  $b$ , the outcome  $y$  is normal with mean

$$E(y|b) = X\beta + Zb$$

and variance  $\Sigma = \sigma^2 I_N$ , where  $N = \sum_i n_i$ . The mixed model framework covers the inverse problems (O'Sullivan 1986) by defining  $Z$  properly. For our current problem we have

$$f_i = \beta + b_i,$$

and, for identifiability, assume that  $E(b_i) = 0$ . Here  $X$  is simply a column of ones of length  $N$ , and  $Z$  is an  $N \times n$  design matrix of zeros and ones;

the row of  $Z$  associated with original data  $(x_i, y_{ij})$  has value one at the  $i$ 'th location and zero otherwise. The random effects  $b$  is simply the mean-corrected version of  $f$ . The actual estimate  $\hat{b}$  depends on the smoothness assumption of the function  $f(x)$ .

It is instructive to see what we get if we simply assume that  $b$  is a fixed effect parameter. The data structure is that of a one-way model

$$y_{ij} = \beta + b_i + e_{ij},$$

where, for identifiability, we typically assume  $\sum_i n_i b_i = 0$ . In this setup the regularity of the grid points  $x_1, \dots, x_n$  is not needed; in fact, we have to drop the  $x_i$  values where  $n_i = 0$ , since for those values  $f(x)$  is not estimable. For simplicity, we just relabel the points for which  $n_i > 0$  to  $x_1, \dots, x_n$ , so we can use the same notation as before. The estimate of  $b_i$  is

$$\hat{b}_i = \bar{y}_i - \bar{y}$$

where  $\bar{y} = \sum_i y_{ij}/N$  is the grand mean and  $\bar{y}_i$  is simply the average of the data of the  $i$ 'th bin, and the estimate of  $f_i$  (regardless of the constraint) is

$$\hat{f}_i = \bar{y}_i. \quad (18.1)$$

The variance of this estimate is  $\sigma^2/n_i$ . If  $n_i$  is small, which is likely if the bin size is small, the statistical noise in this simple formula would be large, obscuring the underlying patterns in the function  $f(x)$ . The purpose of smoothing is to reduce such noise and to reveal the patterns of  $f(x)$ .

### 18.3 Imposing smoothness using random effects model

The assumption about the random effects  $b$  depends on the nature of the function. If  $f(x)$  is smooth, then the smoothness can be expressed by assuming that the differences

$$\Delta b_j = b_j - b_{j-1} \quad (18.2)$$

or the second differences

$$\Delta^2 b_j = b_j - 2b_{j-1} + b_{j-2} \quad (18.3)$$

are iid normal with mean zero and variance  $\sigma_b^2$ . In general we can define differencing of order  $d$  as  $\Delta^d b_j$ , and smoothness can be imposed by assuming that it is iid with some distribution.

For example, assuming  $d = 1$ , we have

$$b_j = b_{j-1} + e_j,$$

where  $e_j$ 's are an iid sequence; this means  $b$  is a first-order random walk on the grid. Figure 18.4 shows some simulated normal random walks of

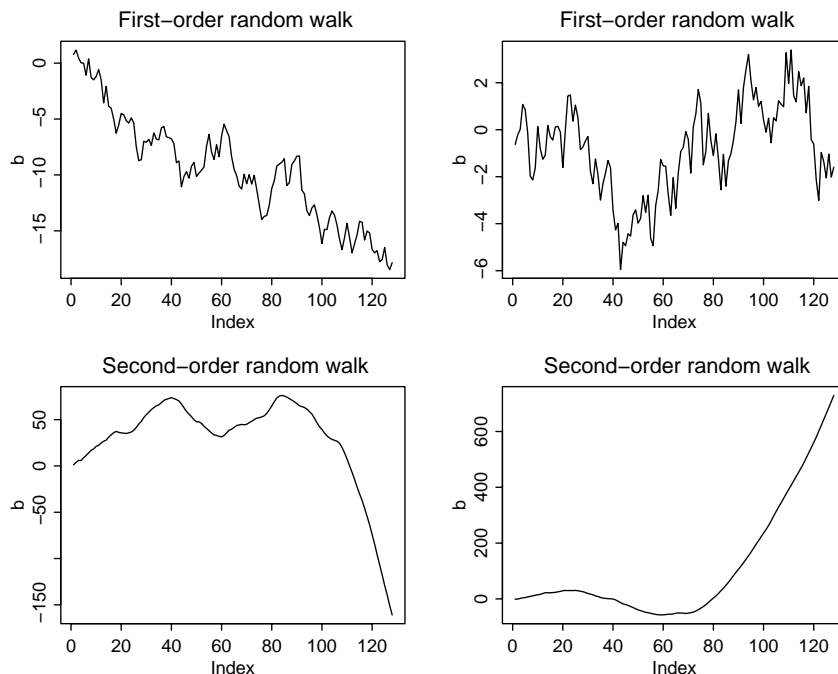


Figure 18.4: *Top row: simulated random walks of the form  $b_j = b_{j-1} + e_j$ , for  $j = 1, \dots, 128$ , where  $b_1 \equiv 0$  and  $e_j$ 's are iid  $N(0, 1)$ . Bottom row:  $b_j = 2b_{j-1} - b_{j-2} + e_j$ , for  $j = 1, \dots, 128$ , where  $b_1 = b_2 \equiv 0$  and  $e_j$ 's are the same as before.*

order 1 and 2; it is clear that the trajectory of random walks of order 2 can mimic a smooth function. The first differencing might be used to allow higher local variation in the function.

Redefining the notation  $\Delta$  for the whole vector

$$\Delta b \equiv \begin{pmatrix} b_2 - b_1 \\ b_3 - b_2 \\ \vdots \\ b_n - b_{n-1} \end{pmatrix}$$

and assuming that  $\Delta b$  is normal with mean zero and variance  $\sigma_b^2 I_{n-1}$ , we have the prior log-likelihood of  $b$  given by

$$\begin{aligned} \log p(b) &= -\frac{n-1}{2} \log \sigma_b^2 - \frac{1}{2\sigma_b^2} b' \Delta' \Delta b \\ &= -\frac{n-1}{2} \log \sigma_b^2 - \frac{1}{2\sigma_b^2} b' R^{-1} b \end{aligned} \quad (18.4)$$

where

$$R^{-1} \equiv \Delta' \Delta = \begin{pmatrix} 1 & -1 & & & 0 \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ 0 & & & -1 & 1 \end{pmatrix}.$$

Or, equivalently, we have assumed that  $b$  is normal with mean zero and inverse covariance matrix

$$D^{-1} \equiv \sigma_b^{-2} R^{-1}.$$

Note that  $\log p(b)$  is a conditional log-likelihood given  $b_1$ ; it is a convenient choice here, since  $b_1$  does not have a stationary distribution. We may also view  $b$  as having a singular normal distribution, with  $D$  not of full rank; this is a consequence of specifying the distribution for only the set of differences. In contrast to the animal breeding application in Section 17.3, specifying  $R^{-1}$  here is more natural than specifying  $R$  (which is defined as the generalized inverse of  $R^{-1}$ ; in practice we never need to compute it). In both applications  $R$  has a similar meaning as a scaled covariance matrix.

Using the second-order assumption that

$$\Delta^2 b \equiv \begin{pmatrix} b_3 - 2b_2 + b_1 \\ b_4 - 2b_3 + b_2 \\ \vdots \\ b_n - 2b_{n-1} + b_{n-2} \end{pmatrix}$$

is normal with mean zero and variance  $\sigma_b^2 I_{n-2}$ , the prior log-likelihood is the same as (18.4) with  $(n-2)$  in the first term rather than  $(n-1)$ , and

$$R^{-1} \equiv (\Delta^2)' \Delta^2 = \begin{pmatrix} 1 & -2 & 1 & & & & 0 \\ -2 & 5 & -4 & 1 & & & \\ 1 & -4 & 6 & -4 & 1 & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & 1 & -4 & 6 & -4 & 1 \\ & & & 1 & -4 & 5 & -2 \\ 0 & & & & 1 & -2 & 1 \end{pmatrix}.$$

## 18.4 Penalized likelihood approach

Combining the likelihood based on the observation vector  $y$  and the random effects  $b$ , and dropping terms not involving the mean parameters  $\beta$  and  $b$ , we obtain

$$\log L = -\frac{1}{2\sigma^2} \sum_{ij} (y_{ij} - \beta - b_i)^2 - \frac{1}{2\sigma_b^2} b' R^{-1} b.$$

The nonnegative quadratic form  $b' R^{-1} b$  is large if  $b$  is rough, so it is common to call the term a roughness penalty and the joint likelihood a ‘penalized

likelihood' (e.g. Green and Silverman 1993). In the normal case, given  $\sigma^2$  and  $\sigma_b^2$ , the estimates of  $\beta$  and  $b$  are the minimizers of a penalized sum of squares

$$\sum_{ij} (y_{ij} - \beta - b_i)^2 + \lambda b' R^{-1} b,$$

where  $\lambda = \sigma^2 / \sigma_b^2$ .

There is a slight difference in the modelling philosophy between the roughness penalty and mixed model approaches. In the former the penalty term is usually chosen for computational convenience, and it is not open to model criticism. The mixed model approach treats the random effects  $b$  as parameters that require some model, and finding an appropriate model is part of the overall modelling of the data. It is understood that a model assumption may or may not be appropriate, and it should be checked with the data. There are two model assumptions associated with the penalty term:

- *The order of differencing.* The penalty approach usually assumes second-order differencing. Deciding what order of differencing to use in a particular situation is a similar problem to specifying the order of nonstationarity of an ARIMA model in time series analysis. It can be easily seen that under- or over-differencing can create a problem of error misspecification. For example, suppose the true model is a first-order random walk

$$\Delta b_j = e_j,$$

where  $e_j$ 's are an iid sequence. The second-order difference is

$$\Delta^2 b_j = \Delta e_j \equiv a_j,$$

so  $a_j$ 's are no longer an iid sequence, but a moving average (MA) of order one. This is a problem since, usually, the standard smoothing model would assume  $a_j$ 's to be iid.

- *Normality.* A quadratic penalty term is equivalent to assuming normality. This is appropriate if  $f(x)$  varies smoothly, but not if  $f(x)$  has jump discontinuities as it would not allow a large change in  $f(x)$ , and it would force the estimate to be smooth. This is where the linear model setup is convenient, since it can be extended easily to deal with this case by using nonnormal mixed models.

## 18.5 Estimate of $f$ given $\sigma^2$ and $\sigma_b^2$

The joint log-likelihood based on the observation vector  $y$  and the random effects  $b$  is

$$\begin{aligned} \log L &= -\frac{1}{2} \log |\Sigma| - \frac{1}{2} (y - X\beta - Zb)' \Sigma^{-1} (y - X\beta - Zb) \\ &\quad - \frac{n-d}{2} \log \sigma_b^2 - \frac{1}{2\sigma_b^2} b' R^{-1} b \end{aligned}$$

where  $d$  is the degree of differencing. Using the assumption  $\Sigma = \sigma^2 I_N$ , given  $\sigma^2$  and  $\sigma_b^2$ , the estimates of  $\beta$  and  $b$  according to the general formula (17.12) are the solution of

$$\begin{pmatrix} X'X & X'Z \\ Z'X & Z'Z + \lambda R^{-1} \end{pmatrix} \begin{pmatrix} \beta \\ b \end{pmatrix} = \begin{pmatrix} X'y \\ Z'y \end{pmatrix}, \quad (18.5)$$

where  $\lambda = \sigma^2/\sigma_b^2$ . We can show that the combined matrix on the left-hand side is singular (Exercise 18.1); it is a consequence of specifying a model only on the set of differences of  $b$ . This implies we can set the level parameter  $\beta$  at an arbitrary value, but by analogy with the fixed effects model it is meaningful to set

$$\hat{\beta} = \bar{y} = \sum_{ij} y_{ij}/N.$$

The estimate of  $b$  is the solution of

$$\{Z'Z + \lambda R^{-1}\}b = Z'(y - X\hat{\beta}). \quad (18.6)$$

From the definition of  $Z$  in this problem, we can simplify (18.6) to

$$(W + \lambda R^{-1})b = W(\bar{y}^v - \bar{y}) \quad (18.7)$$

where  $W = Z'Z = \text{diag}[n_i]$  is a diagonal matrix with  $n_i$  as the diagonal element, and

$$\bar{y}^v \equiv \begin{pmatrix} \bar{y}_1 \\ \vdots \\ \bar{y}_n \end{pmatrix}$$

is the 'raw' mean vector. If  $n_i = 0$  the weight on  $\bar{y}_i$  (which is not available) is zero, so it does not contribute in the computation; we can simply set  $\bar{y}_i$  to zero. (The expression ' $y - \bar{y}$ ' means that the scalar  $\bar{y}$  is subtracted from every element of the vector  $y$ ; this is a common syntax in array-processing computer languages.)

We can also write

$$\begin{aligned} \hat{f} &= \bar{y} + (W + \lambda R^{-1})^{-1}W(\bar{y}^v - \bar{y}) \\ &= (W + \lambda R^{-1})^{-1}W\bar{y}^v + (W + \lambda R^{-1})^{-1}\{(W + \lambda R^{-1})\mathbf{1}_n\bar{y} - W\mathbf{1}_n\bar{y}\} \\ &= (W + \lambda R^{-1})^{-1}W\bar{y}^v + (W + \lambda R^{-1})^{-1}\lambda R^{-1}\mathbf{1}_n\bar{y} \\ &= (W + \lambda R^{-1})^{-1}W\bar{y}^v \end{aligned}$$

since  $R^{-1}\mathbf{1}_n = 0$ , where  $\mathbf{1}_n$  is a vector of ones of length  $n$ .

For the purpose of interpretation, we define a *smoother matrix*  $S_\lambda$  as

$$S_\lambda = (W + \lambda R^{-1})^{-1}W \quad (18.8)$$

so that

$$\hat{f} = S_\lambda \bar{y}^v.$$

We can see that

$$\begin{aligned} S_\lambda \mathbf{1}_n &= (W + \lambda R^{-1})^{-1}W \mathbf{1}_n \\ &= \mathbf{1}_n - (W + \lambda R^{-1})^{-1}\lambda R^{-1} \mathbf{1}_n \\ &= \mathbf{1}_n, \end{aligned}$$

meaning each row of the matrix adds up to one. So we can interpret each  $\hat{f}_i$  as a weighted average of the raw means  $\bar{y}_i$ 's, where the weights are determined by sample size  $n_i$ , the smoothing parameter  $\lambda$  and the choice of  $R^{-1}$ . If the smoothing parameter  $\lambda = 0$  then there is no smoothing, and we are back to the previous estimate (18.1) based on assuming that  $b$  is fixed.

If the data are naturally in a regular grid format such that  $n_i \equiv 1$  for all  $i$ , or we have pairs  $(x_i, y_i)$ , for  $i = 1, \dots, n$ , then  $W = I_n$  and we get

$$\hat{b} = (I_n + \lambda R^{-1})^{-1}(y - \bar{y}),$$

where  $\bar{y} = \sum_i y_i/n$ , and

$$\begin{aligned} \hat{f} &= \bar{y} + (I_n + \lambda R^{-1})^{-1}(y - \bar{y}) \\ &= (I_n + \lambda R^{-1})^{-1}y. \end{aligned}$$

A particular  $\hat{f}_i$  is a weighted average of  $y_i$ 's of the form

$$\hat{f}_i = \sum_j k_{ij} y_j$$

where  $\sum_j k_{ij} = 1$  for all  $i$ . Figure 18.5 shows the shape of the weights  $k_{ij}$ 's for  $i = 1, 10$  and  $20$ , and for  $d = 1$  and  $2$ .

A more 'physical' interpretation of the amount of smoothing can be given in terms of the model *degrees of freedom* or the number of parameters associated with the function estimate. This number of parameters is also useful to make a like-with-like comparison between different smoothers. By analogy with the parametric regression model the degrees of freedom are defined as

$$\text{df} = \text{trace } S_\lambda. \quad (18.9)$$

This is a measure of model complexity: as  $\lambda$  gets larger the estimate becomes more smooth, the degrees of freedom drop, and the estimate gets closer to a parametric estimate. If  $\lambda \rightarrow 0$  we get the number of nonempty bins as the degrees of freedom.

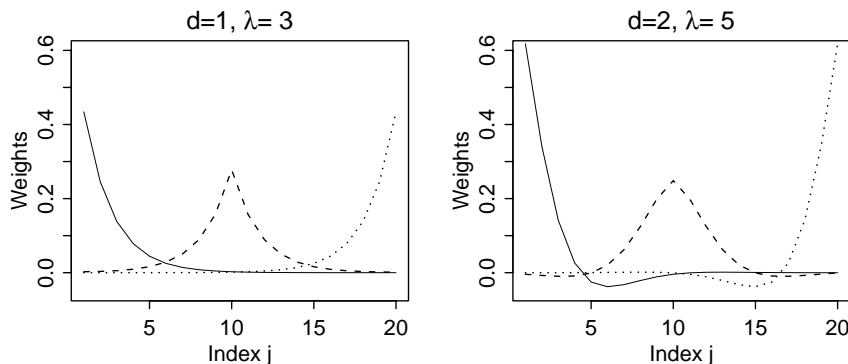


Figure 18.5: The shape of the weights  $k_{ij}$ 's as a function of index  $j$ , for  $i = 1$  (left edge, solid line), 10 (middle location, dashed line) and 20 (right edge, dotted line). The smoothing parameter  $\lambda$  is chosen so both smoothers for  $d = 1$  and  $d = 2$  have around 6 degrees of freedom.

In principle we can compute  $\hat{b}$  by simply solving the linear equation (18.7), but in practice  $n$  may be large, so a simple-minded inversion of the matrix is not efficient. In all large-scale inversion problems we have to exploit the particular structure of the matrix:

- Note that  $R^{-1}$  is a band matrix (with one or two nonzero values on each side of the diagonal). Since  $W$  is a diagonal matrix, the matrix  $(W + \lambda R^{-1})$  is also a band matrix. Finding a very fast solution for such a matrix is a well-solved problem in numerical analysis; see Dongarra *et al.* (1979, Chapter 2) for standard computer programs available in `Linpack` (a collection of programs for linear/matrix computations, such as finding solutions of linear equations).
- The Gauss–Seidel algorithm (Press *et al.* 1992, page 855) works well for this problem.
- If the weights  $n_i$ 's are equal, so that  $W$  is a constant times the identity matrix, then we can use the Fourier transform method (Press *et al.* 1992, Chapters 12 and 13).

(The details of these algorithms are beyond the scope of this text, but serious students of statistics should at some point learn all of these methods.)

**Example 18.2:** We now apply the methodology to the  $\text{SO}_2$  data given in Table 18.1 where  $n = 20$  and  $N = 41$ . The bin statistics are given in Table 18.2, where ‘NA’ means ‘not available’. Figure 18.6 shows the nonparametric smooth of  $\bar{y}^v$  using smoothing parameters  $\lambda = 5$  and  $\lambda = 0.5$ .  $\square$

## 18.6 Estimating the smoothing parameter

Estimating the smoothing parameter  $\lambda = \sigma^2 / \sigma_b^2$  is equivalent to estimating the variance components. We have described before the general problem

Bin $i$	1	2	3	4	5	6	7	8	9	10
$n_i$	1	2	0	1	3	3	0	5	1	3
$\bar{y}_i$	3.43	3.11	NA	3.58	2.94	2.68	NA	2.54	3.26	2.86

Bin $i$	11	12	13	14	15	16	17	18	19	20
$n_i$	8	4	3	3	2	0	0	1	0	1
$\bar{y}_i$	3.45	2.96	2.94	3.23	3.86	NA	NA	4.23	NA	4.7

Table 18.2: Bin statistics for the  $SO_2$  data

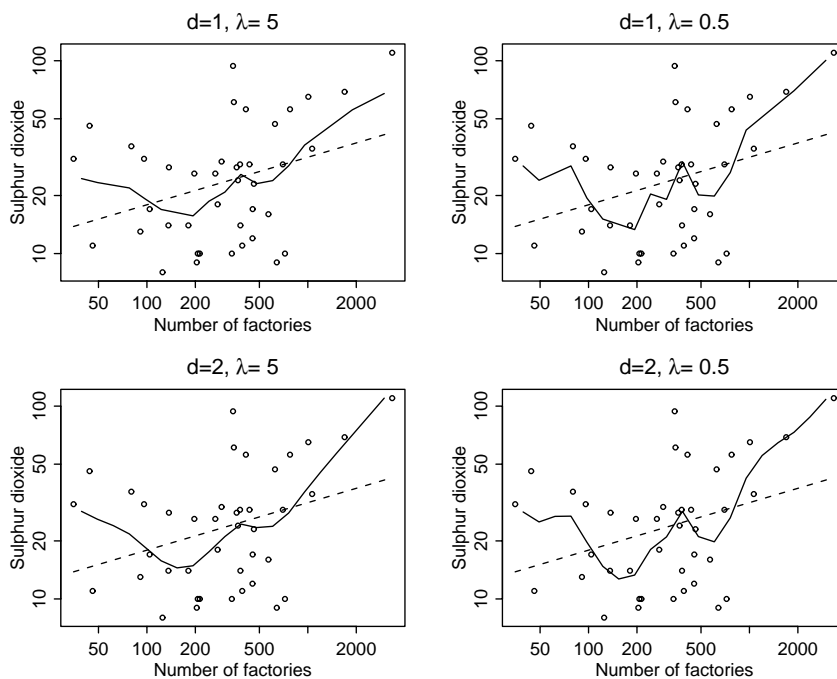


Figure 18.6: Nonparametric smoothing of the  $SO_2$  level against the industrial activity using the mixed model approach. The top row is based on the first-difference assumption and the bottom row on the second-difference. The dashed line on each plot is the linear fit.

of estimating the variance components  $\theta = (\sigma^2, \sigma_b^2)$  using the profile log-likelihood

$$\log L(\theta) = -\frac{1}{2} \log |V| - \frac{1}{2} (y - X\hat{\beta})V^{-1}(y - X\hat{\beta})$$

where  $\hat{\beta}$  is computed according to (17.7), and  $\theta$  enters the function through

$$V = \sigma^2 I_N + \sigma_b^2 ZRZ',$$

or using the equivalent form described in Section 17.5. In this case we want to maximize

$$\begin{aligned}
 Q &= -\frac{N}{2} \log \sigma^2 - \frac{1}{2\sigma^2} (y - X\beta - Zb)' (y - X\beta - Zb) \\
 &\quad - \frac{n-d}{2} \log \sigma_b^2 - \frac{1}{2\sigma_b^2} b' R^{-1} b \\
 &\quad - \frac{1}{2} \log |\sigma^{-2} W + \sigma_b^{-2} R^{-1}|.
 \end{aligned}$$

with respect to all the parameters.

To apply the iterative algorithm in Section 17.5: start with an estimate of  $\sigma^2$  and  $\sigma_b^2$  (note that  $\sigma^2$  is the error variance, so we can get a good starting value for it), then:

1. Compute  $\hat{\beta} = \bar{y}$ , and  $\hat{b}$  according to (18.7), and the error  $e = y - \hat{\beta} - Z\hat{b}$ .
2. Compute the degrees of freedom of the model

$$\text{df} = \text{trace}\{(W + \lambda R^{-1})^{-1} W\},$$

and update  $\theta$  using

$$\begin{aligned}
 \sigma^2 &= \frac{e'e}{N - \text{df}} \\
 \sigma_b^2 &= \frac{1}{n-d} [b' R^{-1} b + \sigma^2 \text{trace}\{(W + \lambda R^{-1})^{-1} R^{-1}\}],
 \end{aligned}$$

where all unknown parameters on the right-hand side are evaluated at the last available values during the iteration, and update  $\lambda = \sigma^2 / \sigma_b^2$ .

Recall that  $d$  is the degree of differencing used for the random effects.

3. Iterate 1 and 2 until convergence.

**Example 18.3:** To apply the algorithm to the SO<sub>2</sub> data we start with  $\sigma^2 = 0.35$  (e.g. use a coarse partition on the data and obtain the error variance) and  $\lambda = 5$  (or  $\sigma_b^2 = 0.35/5$ ). For order of differencing  $d = 1$  the algorithm converges to

$$\begin{aligned}
 \hat{\sigma}^2 &= 0.3679 \\
 \hat{\sigma}_b^2 &= 0.0595
 \end{aligned}$$

with the corresponding smoothing parameter  $\hat{\lambda} = 6.2$  and model degrees of freedom  $\text{df} = 5.35$ . The resulting estimate  $\hat{f}$  is plotted in Figure 18.7. Also shown is the quadratic fit of the data, which has 3 degrees of freedom for the model.

For  $d = 2$ , using the same starting values as above, the algorithm converges to

$$\begin{aligned}
 \hat{\sigma}^2 &= 0.3775 \\
 \hat{\sigma}_b^2 &= 0.0038
 \end{aligned}$$

with the corresponding smoothing parameter  $\hat{\lambda} = 99.2$  and model degrees of freedom  $\text{df} = 3.56$ , very close to the quadratic fit. The estimate using  $d = 2$  is

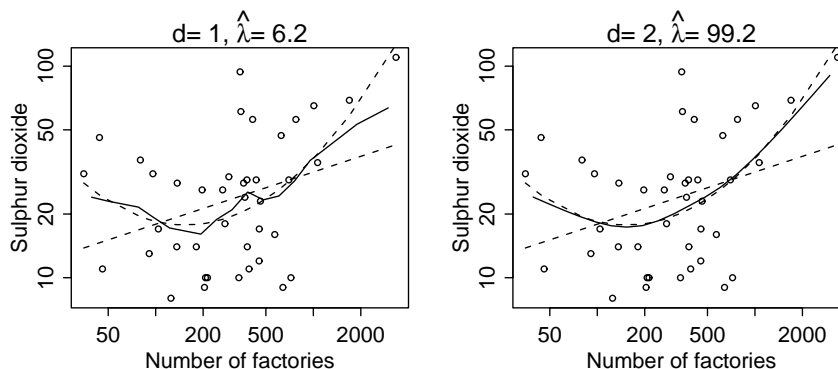


Figure 18.7: *Nonparametric smoothing of the  $SO_2$  level against the industrial activity using the estimated smoothing parameter, with corresponding degrees of freedom  $df = 5.35$  for  $d = 1$ , and  $df = 3.56$  for  $d = 2$ . The dashed lines are linear and quadratic fits of the data.*

more ‘pleasing’, while using  $d = 1$  the estimate appears to show some spurious local patterns. A formal comparison between the fits can be done using the AIC; for the current problem

$$AIC = N \log \hat{\sigma}^2 + 2 \text{ df}.$$

We obtain  $AIC = -30.3$  for  $d = 1$ , and a preferable  $AIC = -32.8$  for  $d = 2$ .  $\square$

### Generalized cross-validation

The generalized cross-validation (GCV) score was introduced by Craven and Wahba (1979) for estimation of the smoothing parameter  $\lambda$  in nonparametric regression. In our setting the score is of the form

$$GCV(\lambda) = \frac{e'e}{(N - df)^2},$$

where the error  $e = y - \hat{\beta} - Z\hat{b}$  and degrees of freedom  $df$  are computed at fixed  $\lambda$ . The estimate  $\hat{\lambda}$  is chosen as the minimizer of the GCV. The justification of the GCV (Wahba 1990, Chapter 4) is beyond the scope of our text.

In some sense  $GCV(\lambda)$  is a profile objective function for  $\lambda$ , which makes the estimation of  $\lambda$  a simple one-dimensional problem. Given  $\hat{\lambda}$  we can estimate the error variance as

$$\hat{\sigma}^2 = \frac{e'e}{(N - df)} \quad (18.10)$$

where  $e$  and  $df$  are computed at  $\hat{\lambda}$ .

Figure 18.8 shows the GCV as a function of  $\lambda$  for the  $SO_2$  data. The minimum is achieved at  $\hat{\lambda} \approx 135$ , with a corresponding degrees of freedom  $df = 3.35$ , very close to the MLE given earlier.

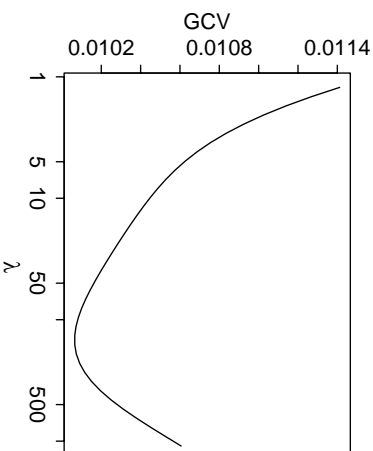


Figure 18.8: The generalized cross-validation (GCV) score as a function of the smoothing parameter  $\lambda$  for the  $\text{SO}_2$  data. The minimum is at  $\hat{\lambda} \approx 135$ .

## 18.7 Prediction intervals

Assuming  $\theta$  is known the Fisher information for  $b$  based on the joint likelihood is given by (17.11). For the current problem

$$I(\hat{b}) = (\sigma^{-2} Z'Z + \sigma_b^{-2} R^{-1}) = \sigma^{-2} (W + \lambda R^{-1}),$$

and the standard errors of the estimates are the square roots of the diagonal elements of

$$I(\hat{b})^{-1} = \sigma^2 (W + \lambda R^{-1})^{-1}.$$

In practice the unknown variance parameters are evaluated at the estimated values. Since  $\hat{\beta}$  is not estimated (recall that it is constrained to the mean value for identifiability reasons),  $\text{se}(\hat{f}_i) = \text{se}(\hat{b}_i)$  and we can construct the 95% prediction interval

$$\hat{f}_i \pm 1.96 \text{se}(\hat{f}_i)$$

for each  $i$ .

Figure 18.9 shows the prediction band for  $f(x)$  in the  $\text{SO}_2$  data. We use the previously estimated values for  $\sigma^2$  and  $\lambda$ . The upper limit is formed by joining the upper points of the prediction intervals, and similarly with the lower limit.

## 18.8 Partial linear models

Suppose we observe independent data  $(x_i, u_i, y_{ij})$ , for  $i = 1, \dots, n$ , where  $x_i$  is a  $p$ -vector of predictors and  $u_i$  is a scalar predictor. A general model of the form

$$E(y_{ij}) = x_i' \beta + f(u_i)$$

and  $\text{var}(y_{ij}) = \sigma^2$  is called a partial linear model (e.g. Speckman 1988). For example, this is used as a generalization of analysis of covariance, where  $\beta$

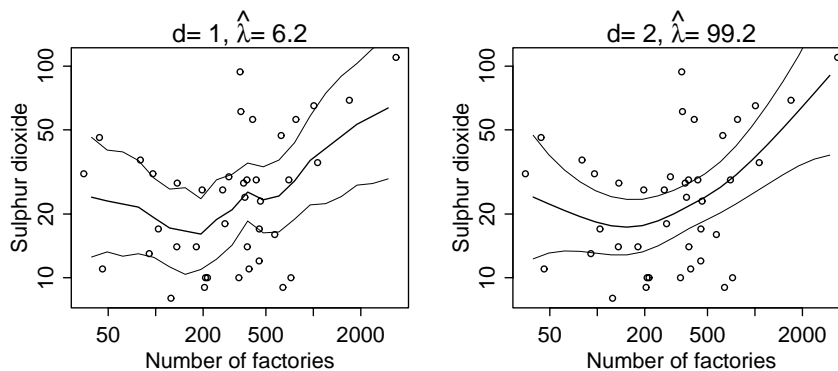


Figure 18.9: *The prediction band for the nonparametric function  $f(x)$  based on pointwise 95% prediction intervals.*

is a measure of treatment effect, and  $f(u_i)$  measures a nonlinear effect of the covariate  $u_i$ .

Assuming that  $u_i$ 's are equispaced and following the previous development, we can write the partial linear model in the form

$$E(y|b) = X\beta + Zb,$$

where  $X$  is an  $N \times p$  design matrix,  $\beta$  is a fixed effects parameter, and  $b$  is a random effects parameter satisfying a smoothness condition. The model is a linear mixed model, and the estimates of  $\beta$  and  $b$  are the solution of

$$\begin{aligned} (X'V^{-1}X)\beta &= X'V^{-1}y \\ (Z'\Sigma^{-1}Z + D^{-1})b &= Z'\Sigma^{-1}y, \end{aligned}$$

where  $V = \Sigma + ZDZ'$ . An iterative backfitting procedure may be used to avoid the computation of  $V^{-1}$ .

## 18.9 Smoothing nonequispaced data\*

Occasionally we face an application where  $x_i$ 's are not equispaced, and we are unwilling to prebin the data into equispaced intervals. The previous methodology still applies with little modification, but with more computations. The problem has a close connection with the general interpolation method in numerical analysis.

It is convenient to introduce the 'design points'  $d_1, \dots, d_p$ , which do not have to coincide with the data points  $x_1, \dots, x_n$ . These design points can be chosen for computational convenience, as with regular grids, or for better approximation, as with the so-called Chebyshev points for the Lagrange polynomial in Example 18.5. We will consider a class of functions defined by

$$f(x) = \sum_{j=1}^p b_j K_j(x),$$

where  $K_j(x)$  is a known function of  $x$  and the design points, and  $b_j$ 's are the parameter values determined by  $f(x)$  at the design points. So, in effect,  $f(x)$  is a linear model with  $K_j(x)$ 's as the predictor variables. In function estimation theory  $K_j(x)$  is called *the basis function*. The nonparametric nature of  $f(x)$  is achieved by allowing  $p$  to be large or, equivalently, by employing a rich set of basis functions.

**Example 18.4:** The simplest example is the power polynomial

$$f(x) = \sum_{j=1}^p b_j x^{j-1},$$

where we have used the basis function

$$K_i(x) = x^{j-1}.$$

Finding  $b_j$ 's is exactly the problem of estimating the regression coefficients in a polynomial regression model; the design points do not play any role in this case. Extending the power polynomial to a high degree is inadvisable because of numerical problems.  $\square$

**Example 18.5:** Using the Lagrange polynomial

$$\begin{aligned} f(x) &= \sum_{j=1}^p f(d_j) \prod_{k \neq j} \frac{x - d_k}{d_j - d_k} \\ &= \sum_{j=1}^p b_j K_j(x) \end{aligned}$$

where  $b_j \equiv f(d_j)$  and

$$K_j(x) \equiv \prod_{k \neq j} \frac{x - d_k}{d_j - d_k}.$$

Each  $K_j(x)$  is a polynomial of degree  $(p-1)$ . The main advantage of the Lagrange polynomial is that the coefficients are trivially available. However, the choice of the design points can make a great difference; in particular, the uniform design is inferior to the Chebyshev design:

$$d_i = \frac{a+b}{2} + \frac{a-b}{2} \cos \frac{(2i-1)\pi}{2p}$$

for  $p$  points between  $a$  and  $b$ .  $\square$

**Example 18.6:** The *B-spline* basis (deBoor 1978) is widely used because of its local properties:  $f(x)$  is determined only by values at neighbouring design points; in contrast, the polynomial schemes are global. The  $j$ 'th B-spline basis function of degree  $m$  is a piecewise polynomial of degree  $m$  in the interval  $(d_j, d_{j+m+1})$ , and zero otherwise. The B-spline of 0 degree is simply the step function with jumps at points  $(d_i, f(d_i))$ . The B-spline of degree 1 is the polygon that connects  $(d_i, f(d_i))$ ; higher-order splines are determined by assuming a smoothness/continuity condition on the derivatives. In practice it is common to use the cubic B-spline to approximate smooth functions (deBoor 1978; O'Sullivan

1987); this third-order spline has a continuous second derivative. The interpolating B-spline is

$$f(x) = \sum_{j=1}^{p-k-1} b_j K_j(x)$$

where  $K_j(x)$ 's are computed based on the design points or 'knots'  $d_1, \dots, d_p$ . See deBoor (1978) for the cubic B-spline formulae.  $\square$

### Methodology

Given observed data  $(x_1, y_1), \dots, (x_n, y_n)$ , where  $E y_i = f(x_i)$ , we can write a familiar regression model

$$\begin{aligned} y &= \sum_{j=1}^p b_j K_j(x) + e \\ &\equiv Zb + e \end{aligned}$$

where the elements of  $Z$  are

$$z_{ij} = K_j(x_i)$$

for some choice of basis function  $K_j(x)$ .

Since  $f(x)$  is available as a continuous function consider a smoothness penalty of the form

$$\lambda \int |f^{(d)}(x)|^2 dx,$$

where  $f^{(d)}(x)$  is the  $d$ 'th derivative of  $f(x)$ . This is a continuous version of the penalty we use in Section 18.3. In view of  $f(x) = \sum_{j=1}^p b_j K_j(x)$  the penalty can be simplified to a familiar form

$$\lambda b' P b,$$

where the  $(i, j)$  element of matrix  $P$  is

$$\int K_i^{(d)}(x) K_j^{(d)}(x) dx.$$

Hence the previous formulae apply, for example

$$\hat{b} = (Z'Z + \lambda P)^{-1} Z' y.$$

## 18.10 Non-Gaussian smoothing

Using the GLMM theory in Section 17.8 we can extend nonparametric smoothing to non-Gaussian data.

**Example 18.7:** Suppose we want to describe surgical mortality rate  $p_i$  as a function of patient's age  $x_i$ . If we do not believe a linear model, or we are at

an exploratory stage in the data analysis, we may consider a model where the outcome  $y_i$  is Bernoulli with probability  $p_i$  and

$$\text{logit } p_i = f(x_i),$$

for some function  $f$ . In this example there could be a temptation to fit a linear model, since it allows us to state something simple about the relationship between patient's age and surgical mortality. There are, however, many applications where such a statement may not be needed. For example, suppose we want to estimate the annual rainfall pattern in a region, and daily rainfall data are available for a 5-year period. Let  $y_i$  be the number of rainy days for the  $i$ 'th day of the year; we can assume that  $y_i$  is binomial(5,  $p_i$ ), and

$$\text{logit } p_i = f(i),$$

where  $f$  is some smooth function. Rather than for analysing a relationship, the purpose of estimating  $f(x)$  in this application is more for a description or a summary.  $\square$

As before, assume that we can arrange or pre-bin the data into regular grids, so our problem is as follows. Given the observations  $(x_i, y_{ij})$  for  $i = 1, \dots, n$  and  $j = 1, \dots, n_i$ , where  $x_i$ 's are equispaced,  $y_{ij}$ 's are independent outcomes from the exponential family model (Section 6.5) of the form

$$\log p(y_{ij}) = \frac{y_{ij}\theta_i - A(\theta_i)}{\phi} + c(y_{ij}, \phi).$$

Let  $\mu_i \equiv E y_{ij}$ , and assume that for a known link function  $h(\cdot)$  we have

$$h(\mu_i) = f(x_i) \equiv f_i,$$

for some unknown smooth function  $f$ .

To put this in the GLMM framework first vectorize the data  $y_{ij}$ 's into an  $N$ -vector  $y$ . Conditional on  $b$ , the outcome  $y$  has mean  $\mu$  and

$$h(\mu) = X\beta + Zb, \tag{18.11}$$

and  $b$  satisfies some smoothness condition stated in Section 18.3. For the simple setup above

$$h(\mu) = f = \beta + b,$$

so  $X$  is a column of ones of length  $N$ , and  $Z$  is an  $N \times n$  design matrix of zeros and ones; the row of  $Z$  associated with original data  $(x_i, y_{ij})$  has value one at the  $i$ 'th location and zero otherwise.

We will treat the general model (18.11) so that the inverse problems are covered, and all of our previous theories for smoothing and GLMM apply. The joint likelihood of  $\beta$ ,  $\theta$  and  $b$  is

$$\log L(\beta, \theta, b) = \log p(y|b) + \log p(b)$$

where  $p(y|b)$  is in the exponential family given above, and  $p(b)$  is the density of  $b$ . The parameter  $\theta$  includes any other parameter in the model, usually the variance or dispersion parameters.

**Estimating  $f$  given  $\theta$** 

We proceed as in Section 17.10, and some results are repeated here for completeness. Given a fixed value of  $\theta$  we use a quadratic approximation of the likelihood to derive the IWLS algorithm; see Section 6.7. Starting with initial values for  $\beta^0$  and  $b^0$ , the exponential family log-likelihood can be approximated by

$$-\frac{1}{2} \log |\Sigma| - \frac{1}{2} (Y - X\beta - Zb)' \Sigma^{-1} (Y - X\beta - Zb), \quad (18.12)$$

where  $Y$  is a working vector with elements

$$Y_i = x_i' \beta^0 + z_i' b^0 + \frac{\partial h}{\partial \mu_i} (y_i - \mu_i^0),$$

and  $\Sigma$  is a diagonal matrix of the variance of the working vector with diagonal elements

$$\Sigma_{ii} = \left( \frac{\partial h}{\partial \mu_i} \right)^2 \phi v_i(\mu_i^0),$$

where  $\phi v_i(\mu_i^0)$  is the conditional variance of  $y_i$  given  $b$ . The derivative  $\partial h / \partial \mu_i$  is evaluated at the current values of  $\beta$  and  $b$ . Alternatively we might use the term ‘weight’  $w_i = \Sigma_{ii}^{-1}$ , and weight matrix  $W = \Sigma^{-1}$ .

If the random effects parameter  $b$  is assumed normal with mean zero and variance  $\sigma_b^2 R$ , where  $R$  is as described in Section 18.3, we have the familiar mixed model equation

$$\begin{pmatrix} X' \Sigma^{-1} X & X' \Sigma^{-1} Z \\ Z' \Sigma^{-1} X & Z' \Sigma^{-1} Z + \sigma_b^{-2} R^{-1} \end{pmatrix} \begin{pmatrix} \beta \\ b \end{pmatrix} = \begin{pmatrix} X' \Sigma^{-1} Y \\ Z' \Sigma^{-1} Y \end{pmatrix} \quad (18.13)$$

to update  $\beta$  and  $b$ . Or, using the iterative backfitting algorithm, we can solve

$$(Z' \Sigma^{-1} Z + \sigma_b^{-2} R^{-1}) b = Z' \Sigma^{-1} (Y - Z\beta)$$

to update  $b$ , and similarly for  $\beta$ . By analogy with the standard regression model the quantity

$$\text{df} = \text{trace}\{(Z' \Sigma^{-1} Z + \sigma_b^{-2} R^{-1})^{-1} Z' \Sigma^{-1} Z\}$$

is called the degrees of freedom associated with  $b$ . The use of nonnormal random effects is described in Section 18.12.

**Example 18.8:** We will now analyse the surgical mortality data in Table 6.2, grouped into 20 bins given in Table 18.3. Let  $y_i = \sum_j y_{ij}$  be the number of deaths in the  $i$ 'th bin, and assume that  $y_i$  is binomial( $n_i, p_i$ ) with dispersion parameter  $\phi = 1$ . We want to estimate  $f$  such that

$$\text{logit } p_i = f_i = \beta + b_i.$$

To use the above methodology, start with  $\beta^0$  and  $b^0 = 0$  and compute the working vector  $Y$  with element

Bin $i$	1	2	3	4	5	6	7	8	9	10
Mid- $x_i$	50.5	51.6	52.6	53.7	54.7	55.8	56.8	57.9	58.9	60.0
$n_i$	3	0	1	3	2	2	4	1	1	2
$\sum_j y_{ij}$	0	NA	0	0	0	0	2	0	0	1
Bin $i$	11	12	13	14	15	16	17	18	19	20
Mid- $x_i$	61.0	62.1	63.1	64.2	65.2	66.3	67.3	68.4	69.4	70.5
$n_i$	4	4	3	2	1	0	2	2	1	2
$\sum_j y_{ij}$	2	3	1	1	0	NA	2	1	0	1

Table 18.3: *Bin statistics for the surgical mortality data in Table 6.2. 'NA' means 'not available'.*

$$Y_i = \beta^0 + b_i^0 + \frac{y_i - n_i p_i^0}{n_i p_i^0 (1 - p_i^0)}$$

and weight  $w_i = \Sigma_{ii}^{-1} = n_i p_i^0 (1 - p_i^0)$ . The matrix  $X$  is a column of ones and  $Z$  is an identity matrix  $I_{20}$ . We then compute the following updates:

$$\begin{aligned}\beta &= \frac{\sum_i w_i (Y - b)}{\sum_i w_i} \\ b &= (W + \sigma_b^{-2} R^{-1})^{-1} W (Y - \beta),\end{aligned}$$

where  $W = \text{diag}[w_i]$ . The iteration continues after recomputing  $Y$  and  $\Sigma$ . So the computation in non-Gaussian smoothing involves an iteration of the Gaussian formula. The model degrees of freedom associated with a choice of  $\sigma_b^2$  are

$$\text{df} = \text{trace}\{(W + \sigma_b^{-2} R^{-1})^{-1} W\}.$$

Figure 18.10 shows the nonparametric smooth of  $p_i$  using smoothing parameter  $\sigma_b^2 = 0.2$  and 2, with the corresponding 4.3 and 6.7 degrees of freedom. The matrix  $R$  used is associated with  $d = 2$ ; see Section 18.3. For comparison the linear logistic regression fit is also shown. The result indicates some nonlinearity in the relationship between age and mortality, where the effect of age appears to flatten after age 62.  $\square$

### Estimating the smoothing parameter

The discussion and method in Section 17.10 for estimating the variance components in GLMM apply here. In general we can choose  $\theta$  to maximize

$$\log L(\theta) = \log L(\hat{\beta}, \theta, \hat{b}) - \frac{1}{2} \log |Z' \Sigma^{-1} Z + D^{-1}|, \quad (18.14)$$

where  $\theta$  enters through  $\Sigma$  and  $D^{-1}$ . This approximate profile likelihood can be maximized using any derivative-free optimization routine.

In the important special case of non-Gaussian outcomes involving a single function estimation, we typically assume  $\phi = 1$ , so  $\theta = \sigma_b^2$ . Since with smooth functions we do not expect  $\sigma_b^2$  to be too large, we can use the following algorithm. Start with an initial estimate of  $\sigma_b^2$ , then:

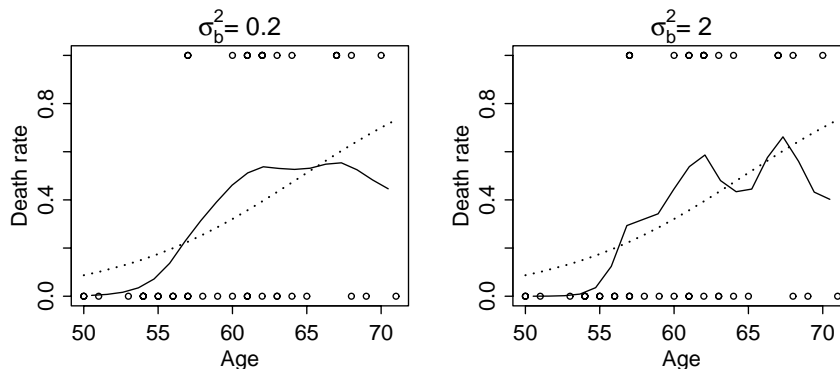


Figure 18.10: *Nonparametric smooth (solid) of mortality as a function of age compared with the linear logistic fit (dotted). The circles are the data points.*

1. Compute  $\hat{\beta}$  and  $\hat{b}$  given  $\sigma_b^2$  according to the method in the previous section.
2. Fixing  $\beta$  and  $b$  at the values  $\hat{\beta}$  and  $\hat{b}$ , update  $\sigma_b^2$  using

$$\sigma_b^2 = \frac{1}{n-d} [b'R^{-1}b + \text{trace}\{(Z'\Sigma^{-1}Z + \sigma_b^{-2}R^{-1})^{-1}R^{-1}\}], \quad (18.15)$$

where  $n$  is the length of  $b$  and  $d$  is the degree of differencing used to define  $R$  (so  $n - d$  is the rank of  $R$ ).

3. Iterate between 1 and 2 until convergence.

This procedure applies immediately to the mortality data example. Figure 18.11(a) shows the mortality rate as a function of age using the estimated  $\hat{\sigma}_b^2 = 0.017$ , with corresponding  $df = 2.9$ .

### Prediction intervals

From Section 18.7, assuming the fixed parameters are known at the estimated values, the Fisher information for  $b$  is

$$I(\hat{b}) = (Z'\Sigma^{-1}Z + \sigma_b^{-2}R^{-1}).$$

We can obtain approximate prediction intervals for  $p_i$  as follows. First obtain the prediction interval for  $f_i$  in the logit scale

$$\hat{f}_i \pm 1.96 \text{ se}(\hat{b}_i),$$

where  $\text{se}(\hat{b}_i)$  is computed from  $I(\hat{b})$  above, then transform the end-points of the intervals to the original probability scale. A prediction band is obtained by joining the endpoints of the intervals. Figure 18.11(b) shows the prediction band for the mortality rate using the estimated  $\hat{\sigma}_b^2 = 0.017$ .

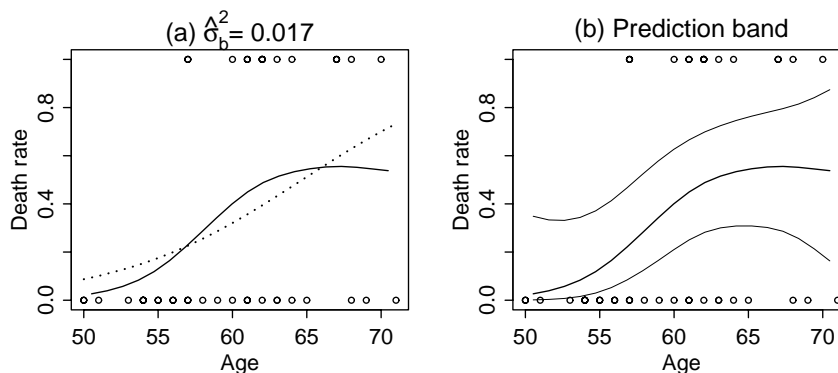


Figure 18.11: (a) Nonparametric smooth (solid) of mortality rate using the estimated  $\hat{\sigma}_b^2 = 0.017$ , compared with the linear logistic fit (dotted). (b) Prediction band for the nonparametric smooth.

## 18.11 Nonparametric density estimation

The simplest probability density estimate is the histogram, a nonparametric estimate based on simple partitioning of the data. When there is enough data the histogram is useful to convey shapes of distributions. The weakness of the histogram is that either it has too much local variability (if the bins are too small), or it has low resolution (if the bins are too large).

The kernel density estimate is commonly used when a histogram is considered too crude. Given data  $x_1, \dots, x_N$ , and kernel  $K(\cdot)$ , the estimate of the density  $f(\cdot)$  at a particular point  $x$  is

$$f(x) = \frac{1}{N\sigma} \sum_i K\left(\frac{x_i - x}{\sigma}\right).$$

$K(\cdot)$  is typically a standard density such as the normal density function; the scale parameter  $\sigma$ , proportional to the ‘bandwidth’ of the kernel, controls the amount of smoothing. There is a large literature on the optimal choice of the bandwidth; see Jones *et al.* (1996) for a review.

**Example 18.9:** Table 12.1 shows the waiting time for  $N = 299$  consecutive eruptions of the Old Faithful geyser in the Yellowstone National Park. The density estimate in Figure 18.12, computed using the Gaussian kernel with  $\sigma = 2.2$ , shows distinct bimodality, a significant feature that indicates nonlinear dynamics in the process that generates it. The choice  $\sigma = 2.2$  is an optimal choice using the unbiased cross-validation score from Scott and Terrell (1987).  $\square$

There are several weaknesses of the kernel density estimate: (i) it is very inefficient computationally for large datasets, (ii) finding the optimal bandwidth (or  $\sigma$  in the above formula) requires special techniques, and (iii) there is an extra bias on the boundaries. These are overcome by the mixed model approach.

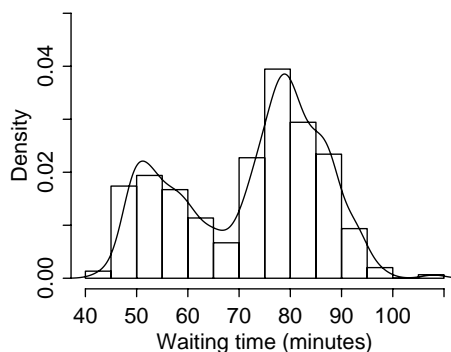


Figure 18.12: *The histogram and kernel density estimate (solid line) of the geyser data indicate strong bimodality.*

First we pre-bin the data, so we have equispaced midpoints  $x_1, \dots, x_n$  with corresponding counts  $y_1, \dots, y_n$ ; there is a total of  $N = \sum_i y_i$  data points. The interval  $\delta$  between points is assumed small enough such that the probability of an outcome in the  $i$ 'th interval is  $f_i \delta$ ; for convenience we set  $\delta \equiv 1$ . The likelihood of  $f = (f_1, \dots, f_n)$  is

$$\log L(f) = \sum_i y_i \log f_i,$$

where  $f$  satisfies  $f_i \geq 0$  and  $\sum_i f_i = 1$ . Using the Lagrange multiplier technique we want an estimate  $f$  that maximizes

$$Q = \sum_i y_i \log f_i + \psi \left( \sum_i f_i - 1 \right).$$

Taking the derivatives with respect to  $f_i$  we obtain

$$\frac{\partial Q}{\partial f_i} = y_i / f_i + \psi.$$

Setting  $\frac{\partial Q}{\partial f_i} = 0$ , so  $\sum f_i (\partial Q / \partial f_i) = 0$ , we find  $\psi = -N$ , hence  $f$  is the maximizer of

$$Q = \sum_i y_i \log f_i - N \left( \sum_i f_i - 1 \right).$$

Defining  $\lambda_i \equiv N f_i$ , the expected number of points in the  $i$ 'th interval, the estimate of  $\lambda = (\lambda_1, \dots, \lambda_N)$  is the maximizer of

$$\sum_i y_i \log \lambda_i - \sum_i \lambda_i,$$

exactly the log-likelihood from Poisson data. We no longer have to worry about the sum-to-one constraint. So, computationally, nonparametric density estimation is equivalent to nonparametric smoothing of Poisson data, and the general method in the previous section applies immediately.

To be specific, we estimate  $\lambda_i$  from, for example, the log-linear model

$$\log \lambda_i = \beta + b_i,$$

where  $b_i$ 's are normal with mean zero and variance  $\sigma_b^2 R$ ; the matrix  $R$  is described in Section 18.3. The density estimate  $\hat{f}_i$  is  $\hat{\lambda}_i/N$ .

### Computing the estimate

Given the smoothing parameter  $\sigma_b^2$ , start with  $\beta^0$  and  $b^0$ , and compute the working vector  $Y$  with element

$$Y_i = \beta^0 + b_i^0 + \frac{y_i - \lambda_i^0}{\lambda_i^0}$$

and weight  $w_i = \Sigma_{ii}^{-1} = \lambda_i^0$ . Update these using

$$\begin{aligned} \beta &= \frac{\sum_i w_i(Y - b)}{\sum_i w_i} \\ b &= (W + \sigma_b^{-2} R^{-1})^{-1} W(Y - \beta), \end{aligned}$$

where  $W = \text{diag}[w_i]$ . In practice we can start with  $b = 0$  and  $\beta^0 = \log \bar{y}$ .

Estimation of  $\sigma_b^2$  is the same as in the previous binomial example; the iterative procedure and updating formula (18.15) for  $\sigma_b^2$  also apply. As before, it is more intuitive to express the amount of smoothing by the model degrees of freedom associated with a choice of  $\sigma_b^2$ :

$$\text{df} = \text{trace}\{(W + \sigma_b^{-2} R^{-1})^{-1} W\}.$$

**Example 18.10:** For the geyser data, first partition the range of the data (from 43 to 108) into 40 intervals. The count data  $y_i$ 's in these intervals are

1 1 2 12 17 5 16 3 11 8 6 8 2 7 2 3 5 11 6 17  
18 17 24 12 14 18 5 21 9 2 11 1 2 1 0 0 0 0 0 1

Figure 18.13 shows the density estimate of the waiting time using the above method (solid line) with  $d = 2$  and an estimated smoothing parameter  $\hat{\sigma}_b^2 = 0.042$  (corresponding  $\text{df} = 11.1$ ). The density estimate matches closely the kernel density estimate using the optimal choice  $\sigma = 2.2$ . □

**Example 18.11:** This is to illustrate the problem of the standard kernel estimate at the boundary. The data are simulated absolute values of the standard normal; the true density is twice the standard normal density on the positive side. The complete dataset is too long to list, but it can be reproduced reasonably using the following information. The values range from 0 to 3.17, and on partitioning them into 40 equispaced intervals, we obtain the following count data  $y_i$ :

17 14 15 20 17 15 16 17 19 14 7 9 14 7 10 11 5 8 5 10  
10 4 6 7 4 2 5 3 1 1 2 0 1 0 0 1 1 0 0 1

Figure 18.14 shows the density estimates using the mixed model approach (solid line, based on  $\hat{\sigma}_b^2 = 0.0006$  or  $\text{df} = 4.3$ ) and the kernel method (dotted line, with optimal choice  $\sigma = 0.06$ ; and dashed line, with  $\sigma = 0.175$ ). Using smaller  $\sigma$  the kernel estimate has less bias at the boundary, but the estimate is visibly too noisy, while larger  $\sigma$  has the opposite problem. □

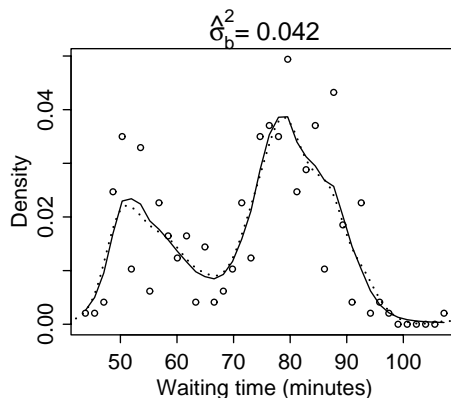


Figure 18.13: The density estimate of the geyser waiting time using the mixed model approach (solid) and the kernel method (dotted). The smoothing parameters of both methods are estimated from the data. The scattered points are the counts  $y_i$ 's scaled so that as a step function they integrate to one.

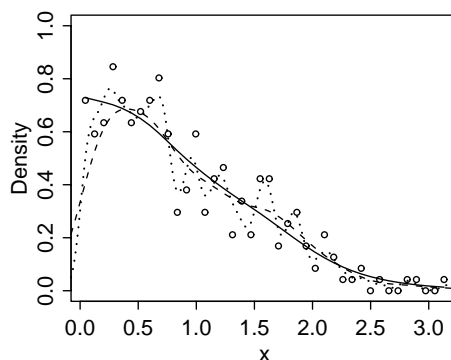


Figure 18.14: The density estimates of a simulated dataset using the mixed model approach (solid) and the kernel method with  $\sigma = 0.06$  (dotted) and  $\sigma = 0.175$  (dashed). The scattered points are the scaled count data.

## 18.12 Nonnormal smoothness condition\*

The normal smoothness assumption is convenient computationally, and it is adequate in most applications. A nonnormal model may be required, for example, if we suspect the underlying function is discontinuous so its derivative might be heavy-tailed. Typically we still make an assumption that the  $d$ 'th-order difference

$$\Delta^d b_i = a_i$$

is iid with some distribution with location zero and scale  $\sigma_b$ ; these do not have to be mean and standard deviation, so, for example, the assumption

covers the Cauchy or double-exponential models.

Let  $\ell(a)$  be the log-likelihood contribution of  $a$ . Using starting value  $b^0$ , as in Section 17.10, we can first approximate  $\ell(a)$  by

$$\ell(a) \approx \ell(a^c) - \frac{1}{2}(a - a^c)'D^{-1}(a - a^c),$$

where  $D^{-1} = \text{diag}[-\ell''(a^0)]$ ,  $a^0 = \Delta^d b^0$ , and

$$a^c = a^0 + D\ell'(a^0).$$

Therefore,

$$\ell(a) \approx \ell(a^c) - \frac{1}{2}(\Delta^d b - a^c)'D^{-1}(\Delta^d b - a^c).$$

The derivative of  $\ell(a)$  with respect to  $b$  is

$$(-\Delta^d)'D^{-1}\Delta^d b + (\Delta^d)'D^{-1}a^c.$$

Combining this with the quadratic approximation of  $\log p(y|b)$ , we obtain the updating equation

$$\begin{pmatrix} X'\Sigma^{-1}X & X'\Sigma^{-1}Z \\ Z'\Sigma^{-1}X & Z'\Sigma^{-1}Z + (\Delta^d)'D^{-1}\Delta^d \end{pmatrix} \begin{pmatrix} \beta \\ b \end{pmatrix} = \begin{pmatrix} X'\Sigma^{-1}Y \\ Z'\Sigma^{-1}Y + (\Delta^d)'D^{-1}a^c \end{pmatrix}.$$

In the normal case,  $a^c = 0$ , and the term  $(\Delta^d)'D^{-1}\Delta^d$  reduces to  $\sigma_b^{-2}R^{-1}$ .

## 18.13 Exercises

**Exercise 18.1:** Show that the combined matrix on the left-hand side of (18.5) is singular.

**Exercise 18.2:** In Example 6.5 find the nonparametric smooth of the number of claims as a function of age. Compare it with the parametric fit. Find the confidence band for the nonparametric fit.

**Exercise 18.3:** Earthquake wave signals exhibit a changing variance, indicating the arrival of the different phases of the wave.

-0.24	-0.19	-0.43	-1.30	-0.16	-1.15	1.42	-0.46	0.85	-0.62
0.12	0.17	-0.32	0.48	-1.38	0.08	-0.22	-1.50	-0.27	2.38
-1.72	-1.14	-0.47	-0.32	2.97	-1.76	-0.36	0.47	-0.89	-5.60
9.30	-3.20	5.42	-7.51	3.44	0.02	-0.29	-9.37	-54.77	4.27
-34.94	26.26	13.51	-87.68	1.85	-13.09	-26.86	-27.29	3.26	-13.75
17.86	-11.87	-11.63	4.55	4.43	-2.22	-56.21	-32.45	12.96	9.80
-6.35	1.17	-2.49	11.47	-7.25	-7.95	-8.03	7.64	25.63	9.12
10.24	-19.08	-3.37	-13.86	7.60	-15.44	5.12	2.90	0.41	-4.92
14.30	5.72	-10.87	1.86	-1.73	-2.53	-1.43	-2.93	-1.68	-0.87
9.32	3.75	3.16	-6.34	-0.92	7.10	2.35	0.24	2.32	-2.72
-2.95	-2.57	-1.63	2.06	-1.66	4.11	0.90	-2.21	2.71	-1.08
-1.22	-0.68	-2.78	-1.91	-2.68	-0.95	1.17	-0.72		

Assume the signal is observed at regular time points  $i = 1, \dots, N$ , and  $y_i$ 's are independent normal with mean zero and variance  $\sigma_i^2$ , where  $\sigma_i^2$  changes smoothly over time.

- (a) Develop a smoothing procedure, including the computational algorithm, to estimate the variance as a function of time.
- (b) Apply it to the observed data.
- (c) Find the prediction band for the variance function.

**Exercise 18.4:** Find the nonparametric estimate of the intensity function from the software failure data in Example 11.9, and compare it with the parametric estimate. Discuss the advantages and disadvantages of each estimate.

**Exercise 18.5:** For the epilepsy data in Section 11.9 find the nonparametric estimate of the baseline intensity function associated with the Cox model as described in Section 11.10.

**Exercise 18.6:** The following time series (read by row) is computed from the daily rainfall data in Valencia, southwest Ireland, from 1985 to 1994. There are 365 values in the series, each representing a calendar date with the 29th of February removed. Each value is the number of times during the ten year period the rain exceeded the average daily amount (4 mm); for example, on January 1st there were 5 times the rain exceeded 4 mm.

```

5 3 6 7 4 2 6 6 7 4 6 3 5 1 2 1 1 5 4 3 4 4 5 5 5 3 2 4 4
3 3 5 3 4 4 2 4 4 5 2 3 4 3 5 5 1 1 4 1 5 2 1 3 3 2 7 4 2
3 4 2 3 4 3 2 2 2 2 1 4 5 3 2 3 1 4 6 2 7 5 2 2 3 1 3 4 4
5 3 5 5 4 4 4 4 2 2 2 2 5 4 1 2 2 3 1 1 2 3 2 2 4 3 4 2 2
1 2 3 2 2 4 2 2 2 1 1 1 1 3 2 1 2 2 4 2 2 1 0 1 2 2 2 2 3
2 1 3 2 2 0 1 4 2 0 4 2 0 3 2 1 3 2 1 1 1 1 3 1 2 3 5 3 0
4 2 5 2 2 2 3 1 3 1 2 3 2 2 1 5 2 4 4 2 3 3 3 3 1 1 0 2
3 3 4 4 3 4 4 1 2 3 1 4 3 4 5 0 3 2 3 5 4 4 3 4 4 3 2 1 4
2 4 4 1 3 3 4 3 2 5 2 1 2 1 2 2 3 2 1 1 3 2 3 1 3 3 1 1 5
3 3 5 1 1 2 3 0 1 3 1 4 6 3 4 4 3 5 5 5 3 3 3 2 2 3 0 1 5
4 5 4 5 2 4 5 5 3 4 5 6 5 2 3 5 2 4 3 2 3 5 8 4 5 5 5 4 4
4 5 4 4 3 3 2 4 4 3 1 3 4 4 3 2 3 5 6 2 5 4 4 1 3 2 3 2 3
4 3 6 2 2 7 5 4 4 7 5 4 5 3 5 6 6

```

Assuming a binomial model for the observed data, compute the smoothed probability of exceeding the mean rainfall as a function of calendar time. Present also the prediction band around the smoothed estimate.